CLAIMS

What is claimed is:

1. A computer implemented method of simulating an integrated circuit design comprising the steps of:

a) accessing an input netlist describing said integrated circuit design, said input netlist organized in a hierarchical fashion;

b) in response to an event, determining a group of leaf cells of said netlist that are effected by said event;

c) of said group of leaf cells, dividing said group into stages based on hierarchical boundaries as defined in said netlist;

d) transforming each of said stages into a separate circuit model;

e) using said circuit models to compute cut node voltages of said stages and recording, in computer memory, said cut node voltages; and

f) repeating said steps b) - e) for multiple events.

2. A method as described in Claim 1 wherein said circuit models are Thevenin equivalent circuit models.

3. A method as described in Claim 1 wherein an event is a change of state of a signal.

4. A method as described in Claim 1 wherein said step d) further comprises the step of generating a sensitivity vector for a stage.

5. A method as described in Claim 4 wherein said step e) comprises the step of computing internal node voltages of a stage based on said sensitivity vector for said stage and said cut node voltages of said stage.

6. A method as described in Claim 1 wherein said step b) reuses any previously computed static information pertinent to instances of a same cell in order to reduce memory consumption.

7. A method as described in Claim 1 wherein said step d) reuses any previously computed circuit model information pertinent to instances of a same cell in order to reduce memory consumption.

8. A method as described in Claim 1 wherein connectivity information describes connections between cells of said input netlist and wherein said step b) utilizes said connectivity information to determine said group of leaf cells.

9. A computer implemented method of simulating an integrated circuit design comprising the steps of:

a) accessing an input netlist describing said integrated circuit design, said input netlist organized in a hierarchical fashion of cells and comprising connectivity information that describes connections between said cells;

b) in response to an event, using said connectivity information to determine a group of leaf cells of said netlist that are effected by said event;

c) of said group of leaf cells, dividing said group into stages based on hierarchical boundaries as defined in said netlist;

d) transforming each of said stages into a separate Thevenin equivalent circuit model;

e) using said Thevenin equivalent circuit models to compute cut node voltages of said stages and recording, in computer memory, said cut node voltages into a flat node voltage data structure; and

f) repeating said steps b) - e) for multiple events.

10. A method as described in Claim 9 wherein an event is a change of state of a signal.

11. A method as described in Claim 9 wherein said step d) further comprises the step of generating a sensitivity vector for a stage.

12. A method as described in Claim 11 wherein said step e) comprises the steps of:

computing internal node voltages of a stage based on said sensitivity vector for said stage and said cut node voltages of said stage; and

storing said internal node voltages into an instance specific dynamic data structure.

13. A method as described in Claim 9 wherein said step b) reuses any previously computed static information pertinent to instances of a same cell in order to reduce memory consumption.

14. A method as described in Claim 9 wherein said step d) reuses any previously computed Thevenin equivalent circuit model information pertinent to instances of a same cell in order to reduce memory consumption.

15. A computer system comprising a processor coupled to bus and a memory coupled to said bus, wherein said memory contains instructions that when executed implement a method of simulating an integrated circuit design comprising the steps of:

a) accessing an input netlist describing said integrated circuit design, said input netlist organized in a hierarchical fashion;

b) in response to an event, determining a group of leaf cells of said netlist that are effected by said event;

c) of said group of leaf cells, dividing said group into stages based on hierarchical boundaries as defined in said netlist;

d) transforming each of said stages into a separate circuit model;

e) using said circuit models to compute cut node voltages of said stages and recording, in computer memory, said cut node voltages; and

f) repeating said steps b) - e) for multiple events.

16. A computer system as described in Claim 15 wherein said circuit models are Thevenin equivalent circuit models.

17. A computer system as described in Claim 15 wherein an event is a change of state of a signal.

18. A computer system as described in Claim 15 wherein said step d) of said method further comprises the step of generating a sensitivity vector for a stage.

19. A computer system as described in Claim 18 wherein said step e) of said method comprises the step of computing internal node voltages of a stage based on said sensitivity vector for said stage and said cut node voltages of said stage.

20. A computer system as described in Claim 18 wherein said step e) of said method comprises the steps of:

computing internal node voltages of a stage based on said sensitivity vector for said stage and said cut node voltages of said stage; and

storing said internal node voltages into an instance specific dynamic data structure.

21. A computer system as described in Claim 15 wherein said step b) of said method reuses any previously computed static information pertinent to instances of a same cell in order to reduce memory consumption.

22. A computer system as described in Claim 15 wherein said step d) of said method reuses any previously computed circuit model information pertinent to instances of a same cell in order to reduce memory consumption.

23. A computer system as described in Claim 15 wherein connectivity information describes connections between cells of said input netlist and wherein step b) utilizes said connectivity information to determine said group of leaf cells.